Menghitung Luas & Keliling Lingkaran Menggunakan Konsep PEWARISAN (INHERITANCE)

Nurlita Abelianti[1],Salsabila Zetia Rahmadani[2],Yuniana Cahyaningrum[3]

Prodi Pendidikan Teknologi Informasi

IKIP PGRI BOJONEGORO

@liaabel458.gmail.com,@Salsabillazetyaramadhani.gmail.com,

yuniana@ikippgribojonegoro.ac.id

ABSTRAK

OOP kepanjangan dari Object Oriented Programming atau biasa disebut dengan Pemrograman Berorientasi Objek. Paradigma pada pemrograman yang berorientasi objek didasari oleh istilah object (Objek) dan class (Kelas). Semua data dan fungsi di dalam paradigma ini dibungkus dalam kelas-kelas atau objek-objek. Bandingkan dengan logika pemrograman terstruktur. Setiap objek dapat menerima pesan, memproses data, dan mengirim pesan ke objek lainnya. Pada jaman sekarang,banyak bahasa pemrograman yang mendukung OOP.

Sebagai Informasi bahwa Ciri khas Pemrograman Berorientasi Objek salah satunya yaitu Inheritance (Pewarisan). Pewarisan adalah salah satu dari tiga konsep dasar OOP. Draf Warisan itu mencakup dunia nyata di mana entitas/objek bisa ada anak makhluk/benda. Konsep pewarisan memungkinkan suatu kelas yang memiliki kelas turunan disebut induk atau kelas dasar. Meskipun kelas turunannya sendiri sering disebut sebagai subclass atau kelas anak. Di dunia nyata, entitas turunan dapat mewarisi apapun yang dimilikinya pria tua Misalnya antara unit ibu dan unit anak. Pewarisan sendiri memiliki beberapa istilah yaitu,Extends (Keyword ini harus kita tambahkan pada definisi class yang menjadi subclass),Superclass (digunakan untuk menunjukkan hirarki class yang berarti class dasar dari subclass/class anak),Subclass (adalah class anak atau turunan secara hirarki dari superclass. Super Keyword ini digunakan untuk memanggil konstruktor dari superclass atau menjadi variabel yang mengacu pada superclass), Methode Overriding (Pendefinisian ulang method yang sama pada subclass).

Kata kunci: Pengertian inheritance,Deklarasi Inheritance,Penerapan Inherintace ,Istilah-istilah Inheritance

ABSTRACT

OOP stands for Object Oriented Programming or commonly known as Object Oriented Programming. The paradigm in object-oriented programming is based on the terms object (Object) and class (Class). All data and functions in this paradigm are wrapped in classes or objects. Compare with structured programming logic. Each object can receive messages, process data, and send messages to other objects. Today, many programming languages support OOP.

As information that one of the characteristics of object-oriented programming is inheritance. Inheritance is one of the three basic OOP concepts. That draft Inheritance includes the real world where entities/objects can have children of that creature/object. The concept of inheritance allows a class that has a derived class to be called the parent or base class. Although the derived class itself is often referred to as a subclass or child class. In the real world, a child entity can inherit whatever belongs to the parent. For example, between a mother unit and a child unit. Inheritance itself has several terms, namely, Extends (this keyword must be added to the definition of a class that becomes a subclass), Superclass (used to indicate a class hierarchy which means the base class of a subclass/child class), Subclass (is a child class or a hierarchical derivative of superclass. This super keyword is used to call the constructor of the superclass or to be a variable that refers to the superclass), Methode Overriding (Redefining the same method in the subclass).

Keywords: Definition of inheritance, Declaration of Inheritance, Application of Inheritance,Terms of Inheritance

PENDAHULUAN

OOP kepanjangan dari Object Oriented Programming atau biasa disebut dengan Pemrograman Berorientasi Objek . Paradigma pada pemrograman yang berorientasi objek didasari oleh istilah object (Objek) dan class (Kelas). Semua data dan fungsi di dalam paradigma ini dibungkus dalam kelas-kelas atau objek-objek.

Bandingkan dengan logika pemrograman terstruktur. Setiap objek dapat menerimapesan, memproses data, dan mengirim pesan ke objek lainnya. Pada jaman sekarang,banyak bahasa pemrograman yang mendukung OOP.

OOP adalah paradigma pemrograman yang cukup dominan saat ini, karena mampu memberikan solusi kaidah pemrograman modern. Meskipun demikian, bukan berarti bahwa pemrograman prosedural sudah tidak layak lagi. Salah satu dari tiga konsep dasar OOP yaitu Inheritance.

Konsep inheritance ini mengadopsi dunia riil dimana suatu entitas/obyek dapat mempunyai entitas/obyek turunan. Dengan konsep inheritance, sebuah class dapat mempunyai class turunan.

TINJAUAN PUSTAKA

A. PENGERTIAN PEWARISAN INHERITANCE

Pemahaman dasar tentang pewarisan. Pewarisan adalah salah satu dari tiga konsep dasar OOP. Draf Warisan itu mencakup dunia nyata di mana entitas/objek bisa ada anak makhluk/benda. Konsep pewarisan memungkinkan suatu kelas yang memiliki kelas turunan disebut induk atau kelas dasar. Meskipun kelas turunannya sendiri sering disebut sebagai subclass atau kelas anak. Di dunia nyata, entitas turunan dapat mewarisi apapun yang dimilikinya pria tua Misalnya antara unit ibu dan unit anak. Unit anak-anak mungkin dikenakan biaya itu semua adalah bagian dari sifat ayahnya. Hal yang sama berlaku untuk konsep warisan. Subclass dapat mewarisi apa saja dari kelas induk. Itulah bagian terpenting dari konsep pewarisan. Ini karena subclass dapat mewarisi semuanya dari superclass-nya anggota subclass terdiri dari properti dan propertinya yang diwarisi dari kelas induknya. Fitur pewarisan pada OOP mengijinkan sebuah kelas dengan tipe superclass untuk menurunkan atribut-atribut dan methodnya kepada kelas yang lainnya yang disebut dengan subclass atau kelas turunannya. Hal tersebut akan memungkinkan pembuatan kelas baru yang didasarkan dari pengabstrakan atribut-atribut dan behaviour yang sama.

1. **OBJECT (OBJEK)**

   Pada dasarnya semua benda yang ada di dunia nyata dapat dianggap sebagai sebuah objek, nah Jika perhatikan lebih lanjut, pada dasarnya ada dua karakteristik yang utama pada sebuah objek , yaitu setiap objek memiliki atribut sebagai status yang kemudian akan disebut sebagai state. Dan Setiap objek memiliki tingkah laku yang kemudian akan disebut sebagai behaviour. Sebagai contoh sederhana adalah : Manusia memiliki state: umur, tinggi, berat dan sebagainya. Demikian pula manusia memiliki behaviour: menua, meninggi, makan (menambah berat badan) dan sebagainya. Dalam pemrograman JAVA state bisanya dikatakan sebagai variabel dan behaviour adalah method.

2. **CLASS (KELAS)**

   Class merupakan prototipe yang mendefinisikan variabel-variabel dan method-method secara umum. Sedangkan objek pada sisi yang lain merupakan instansiasi dari suatu kelas.

3. **ATTRIBUTE**

   Atribut adalah data yang dapat membedakan antara satu obyek dengan obyek yang lain. Contoh untuk class mahasiswa terdapat obyek mahasiswa si A,danobyek mahasiswa si B. Yang membedakan antara obyek si A dan obyek si B adalah NIM-nya yang merupakan Atribut dari obyek tersebut.

B. DEKLARASI PEWARISAN (INHERITANCE)

Deklarasi yang digunakan adalah dengan menambahkan kata kunci extends setelah deklarasi nama class, kemudian diikuti dengan nama parent class-nya. Kata kunci extends tersebut memberitahu kompiler Java bahwa kita ingin melakukan perluasan class. public class B extends A { … }. Pada saat dikompilasi, Kompiler Java akan membacanya sebagai subclass dari class Object. public class A extends Object { … }

C. KAPAN KITA MENERAPKAN INHERITANCE?

Kita baru perlu menerapkan inheritance pada saat kita jumpai ada suatu class yang dapat diperluas dari class lain. Misal terdapat class Pegawai public class Pegawai { public String nama; public double gaji; }Misal terdapat class Manager public class Manajer { public String nama; public double gaji;public String departemen; }

Dari 2 buah class diatas, kita lihat class Manajer mempunyai data member yang identic sama dengan class Pegawai, hanya saja ada tambahan data member departemen. Sebenarnya yang terjadi disana adalah class Manajer merupakan perluasan dari class Pegawai dengan tambahan data member departemen. Disini perlu memakai konsep inheritance, sehingga class Manajer dapat kita tuliskan seperti berikut : public class Manajer extends Pegawai { public String departemen; }

D. ISTILAH INHERITANCE

Extends, Keyword ini harus kita tambahkan pada definisi class yang menjadi subclass. Superclass, Superclass digunakan untuk menunjukkan hirarki class yang berarti class dasar dari subclass/class anak. Subclass, Subclass adalah class anak atau turunan secara hirarki dari superclass. Super Keyword ini digunakan untuk memanggil konstruktor dari superclass atau menjadi variabel yang mengacu pada superclass. Methode Overriding Pendefinisian ulang method yang sama pada subclass.

Dalam inheritance, method overriding berbeda dengan method overloading. Kalau method overriding adalah mendefinisikan kembali method yang sama, baik nama method maupun signature atau parameter yang diperlukan dalam subclass, kalau method overloading adalah mendefinisikan method yang memiliki nama yang sama, tetapi dengan signature yang berbeda dalam definisi class yang sama.

METODE

Metode yang digunakan adalah dengan cara melalui dari beberapa sumber seperti artikel,website serta jurnal.

PEMBAHASAN

```java
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package bangundatar;

/**
 *
 * @author ASUS
 */
class BangunDatar {

    /**
     * @param args the command line arguments
     */
    float luas () {
        System.out.println ("Hitung luas");
        return 0;
    }
    float keliling () {
        System.out.println("Hitung keliling");
        return 0;
    }
}

class Lingkaran extends BangunDatar {
    int r;
```

Gambar 1.1 Method overriding (Induk class)

Pada gambar diatas terdapat package bangundatar yang didalamnya terdapat class BangunDatar() sebagai induk class yang memiliki dua method yaitu method luas dan keliling. Kedua method tersebut akan digunakan juga untuk subclass.

```java
}

class Lingkaran extends BangunDatar {
    int r;

    public Lingkaran (int r) {
        this.r = r;
    }
    @Override
    public float luas () {
        return (float) (Math.PI*r*r);
    }
    @Override
    public float keliling () {
        return (float) (2*Math.PI*r);
    }
}
public class MTK {
    public static void main (String [] args) {
        BangunDatar bd = new BangunDatar();
        Lingkaran l = new Lingkaran(8);
        bd.luas();
        bd.keliling();

        System.out.println ("Luas Lingkaran="+l.luas());
        System.out.println ("Luas Lingkaran="+l.keliling());
    }
}
```

Gambar 1.2 Method overriding (Subclass dan Main Class)

Pada gambar 1.2 di perlihatkan disana terdapat *class Lingkaran extends BangunDatar()* yang artinya class Lingkaran merupakan bagian dari class BangunDatar() atau bisa disebut juga dengan Subclass dari class BangunDatar(). Class Lingkaran mempunyai parameter *int r* yang akan diisi nilainya di Main class nanti.

```
Output - BangunDatar (run)
run:
Hitung luas
Hitung keliling
Luas Lingkaran=201.06194
Luas Lingkaran=50.265484
BUILD SUCCESSFUL (total time: 0 seconds)
```

Gambar 1.3 Method overriding

Pada Main class terdapat objek baru untuk memanggil method luas dan keliling (BangunDatar bd = new BangunDatar()), bd merupakan objek baru tersebut untuk memanggil method luas dan keliling (bd.luas() dan bd.keliling()). Dan juga objek baru l (Lingkaran l = new Lingkaran(8)) yang digunakan untuk memanggil method luas dan keliling juga, dan untuk memberikan nilai pada parameter di class Lingkaran.

OUTPUT:

Hitung luas dan Hitung keliling merupakan hasil pemanggilan dari method luas dan keliling (bd.luas() dan bd.keliling()).

Luas Lingkaran = 201.06194 dan Luas Keliling = 50.265484 merupakan hasil pemanggilan method luas keliling (l.luas() dan l.keliling()).

KESIMPULAN

Pewarisan adalah salah satu konsep dasar Pemrograman Berorientasi Objek atau Object Oriented Programming (OOP). Ini adalah konsep di mana kelas mewarisi properti dan metode dari kelas lain. Kelas yang mewarisi dari kelas lain disebut subclass atau kelas turunan, sedangkan kelas yang diwarisi disebut superclass atau kelas dasar. Pewarisan memungkinkan subclass untuk menggunakan dan menggunakan kembali kode yang telah didefinisikan dalam superclass. Subkelas juga dapat menambahkan properti dan metodenya sendiri, dan mengganti yang diwariskan jika diperlukan. Ini membuat kode lebih modular dan dapat digunakan kembali, serta lebih mudah dipelihara. Pewarisan dideklarasikan dalam subclass dengan menggunakan kata kunci "extends" diikuti dengan nama superclass. Superclass harus didefinisikan sebelum subclass, dan subclass hanya dapat mewarisi dari satu superclass dalam satu waktu. Ada beberapa istilah yang berkaitan dengan pewarisan, seperti: - Superclass: kelas yang diwariskan - Subclass: kelas yang mewarisi dari superclass - Method Overriding : proses pendefinisian ulang metode pada subclass yang sudah didefinisikan pada superclass - Kata Kunci Super: referensi ke superclass, digunakan untuk memanggil konstruktor atau metode superclass dari subclass. Keuntungan pewarisan termasuk penggunaan kembali kode, modularitas, dan pemeliharaan yang lebih mudah. Ini juga memungkinkan pembuatan kelas yang lebih terspesialisasi yang mewarisi properti dan metode dari superclass yang lebih umum.

DAFTAR PUSTAKA

Alza.2022. ***OOP Praktis – Inheritance/pewarisan (python)***: https://koding.alza.web.id/oop-praktis-inheritance-pewarisan/, diakses pada tanggal 02 Mei 2023

Wijayanto,Ardhi.2022. ***Contoh Inheritance Pewarisan di Java:*** https://www.ardhi.web.id/2019/03/contoh-inheritance-pewarisan-di-java.html , diakses pada tanggal 28 April 2023

Yuliana,Mike. Tanpa Tahun. ***Pewarisan (Inheritance)*** : https://mieke.lecturer.pens.ac.id/Dasar%20Pemrograman%202/teorijava/T4-pewarisan%20%5BCompatibility%20Mode%5D.pdf, diakses pada tanggal 29 April 2023